[0017] Fig. 1a shows a symbolic representation of an embodiment of a data structure representing a version designator, according to the present invention.

[0018] Fig. 1b shows an embodiment of a version designator according to the present invention, showing a two level representation of revision numbers for a main object and a sub-object.

[0019] Fig. 2a shows an example of the creation of objects and sub-objects, according to the present invention in the context of a build process flow for a representative mobile phone.

[0020] Fig. 2b shows a process flow for building a mobile phone and an embodiment of a versioning method for programs having objects and sub-objects, according to the present invention.

[0021] Fig. 2c shows a process flow for building a mobile phone, along with the effect of changes in sub-recipes on an embodiment of a version designator of a main recipe, according to the present invention.

[0022] Fig. 3 is a chart showing an embodiment of a version tracking approach, according to the present invention, on a single manufacturing line.

[0023] Fig. 4 is a chart showing an embodiment of a version tracking approach, according to the present invention, on a multiple-line factory.

[0024] Fig. 5 is a schematic representation of a computer-controlled manufacturing system on a network communicating with a data base, a server, and a process engineer, according to the present invention.

DETAILED DESCRIPTION

[0025] Fig. 1a shows a data structure designator that is used for identifying the version, and the status of that version, of a program comprising a main object and sub-objects that together represent an article of manufacture, as well as a manufacturing process step associated with that article. The first character, which in the illustrated embodiment is shown as a generic "X" (01), is a version label and can be a V, the first letter of the word "version." A top-level version identifier (02), after the version label, indicates the number of times the main object was changed, i.e., the revision number of the main object, while a lower-level version identifier after the decimal point represents the revision number of the sub-object(s). Generic letter "Y" after the numbers indicates the status of that particular version. Thus, using as an example a printed circuit board that is processed in an electronic components manufacturing line, the symbol in Fig. 1b represents a second version of the board with a third revision of a sub-object component on that board, and that V2.03R is the released version where "R" stands for "released". Had it been a

validated version, which will be described in greater detail below, the symbol would have been indicated as V2.03V. It will be understood by those skilled in the art that, in general, such a symbol as in Figs. 1a and 1b can represent the status and the revision of any main portion (e.g., object), and sub-portions (e.g., sub-objects) of a program. Also, in this embodiment of the versioning method (although not in general), the version number has at least two increasing numbers straddling a decimal point that separates them. The first number, on the left of the decimal point, represents the revision number of the main object, while the second number represents the revision number of the modification of any of the sub-objects that are associated with the main object.

[0026] Assignment of version numbers to objects, such as boards, starts with initial programming. Thus, Fig. 2a is a schematic drawing showing the initial steps of programming a computer-operated system for the assembly of a board for a mobile telephone. The board may be a multi-layered board with the required wiring layers, and provided with sites on it where other components, such as, resistors, capacitors, transistors, integrated chips and small outline transistors (SOTs), and so forth, are to be joined to the surface of the board using technologies such as solder ball joining, wire bonding and other technologies known in the art. It will also be understood that these various components are fed into placement machines through respective feeders. Furthermore, the system is to be programmed for one specific manufacturing line. Thus, the computer-controlled system is to be